

VŠB – Technická univerzita Ostrava

Fakulta strojní

Katedra automatizační techniky a řízení

## **Sledování nosnic pomocí RFID**

Monitoring of Laying Hens by RFID

Student:

Jan Kopřiva

Vedoucí bakalářské práce:

Ing. David Fojtík, Ph.D.

## Zadání bakalářské práce

Student: **Jan Kopřiva**  
Studijní program: **B2341 Strojírenství**  
Studijní obor: **3902R001 Aplikovaná informatika a řízení**  
Téma: **Sledování nosnic pomocí RFID  
Monitoring of Laying Hens by RFID.**  
Jazyk vypracování: **čeština**

Zásady pro vypracování:

1. Zpracujte rešerši dostupných technologií inteligentního kurníku pro malochov.
2. Seznamte se s technologií RFID a zvolte vhodné hardwarové a softwarové prostředky pro realizaci úlohy sledování nosnic.
3. Navrhněte a vytvořte monitorovací systém nosnic za účely sledování produkce vajec jednotlivých nosnic a obsazenosti kurníku pro ovládání automatických dvířek kurníku.
4. Zhodnoťte dosažené výsledky a navrhněte směry dalšího řešení.

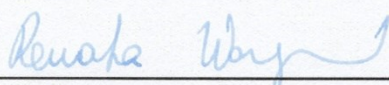
Seznam doporučené odborné literatury:

LEHPAMER, Hrvoj, 2012. *RFID design principles*. 2nd ed. Boston: Artech House, c2012. ISBN 978-1-60807-470-9.  
LÜTTWITZ, Michael von, 2019. *Slepice*. Praha : Vašut. Jak na to (Jan Vašut). ISBN 978-80-7541-166-2.  
MANN, Burkhard, 2003. *C pro mikrokontroléry: ANSI-C, kompilátory C, spojovací programy - linkery, práce s ATMEGA AVR a MSC-51, příklady programování v jazyce C, nástroje pro programování, tipy a triky...* Praha: BEN - technická literatura, 2003.  $\mu$ C & praxe. ISBN 80-7300-077-6.  
SEAL, David, 2001. *ARM architecture reference manual*. 2nd ed. Harlow: Addison-Wesley, 2001, 1 sv. (různé stránkování). ISBN 0-201-73719-1.  
VÁŇA, Vladimír, 2009. *ARM pro začátečníky*. 1. vyd. Praha: BEN - technická literatura, 2009, 195 s. ISBN 978-80-7300-246-6.  
VYSTAVĚL, Radek, 2009, *Moderní programování: učebnice pro začátečníky*. 3. vyd. Ondřejov : moderníProgramování. ISBN 978-80-903951-6-9.


Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. David Fojtík, Ph.D.**

Datum zadání: 20.12.2019  
Datum odevzdání: 18.05.2020

  
doc. Ing. Renata Wagnerová, Ph.D.  
vedoucí katedry



  
prof. Ing. Ivo Hlavatý, Ph.D.  
děkan fakulty

Místopřísežné prohlášení studenta

Prohlašuji, že jsem celou bakalářskou práci včetně příloh vypracoval samostatně pod vedením vedoucího bakalářské práce a uvedl jsem všechny použité podklady a literaturu.

V Ostravě dne 18. 5. 2020

A handwritten signature in blue ink, appearing to read 'Kopru', written over a horizontal dotted line.

Podpis studenta



Prohlašuji, že:

- jsem si vědom, že na tuto moji závěrečnou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. Zákon o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (dále jen Autorský zákon), zejména § 35 (Užití díla v rámci občanských či náboženských obřadů nebo v rámci úředních akcí pořádaných orgány veřejné správy, v rámci školních představení a užití díla školního) a § 60 (Školní dílo),
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen „VŠB-TUO“) má právo užít tuto závěrečnou bakalářskou\*) práci nekomerčně ke své vnitřní potřebě (§ 35 odst. 3 Autorského zákona),
- bude-li požadováno, jeden výtisk této bakalářské práce bude uložen u vedoucího práce,
- s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 Autorského zákona,
- užít toto své dílo, nebo poskytnout licenci k jejímu využití, mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše),
- beru na vědomí, že – podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů - že tato bakalářská\*) práce bude před obhajobou zveřejněna na pracovišti vedoucího práce, a v elektronické podobě uložena a po obhajobě zveřejněna v Ústřední knihovně VŠB-TUO, a to bez ohledu na výsledek její obhajoby.

V Ostravě dne 18. 5. 2020



Podpis studenta

Jméno a příjmení autora práce: Jan Kopřiva

Adresa trvalého pobytu autora práce: Koldín 87

Choceň 565 01

## **Poděkování**

Rád bych poděkoval svému vedoucímu bakalářské práce Ing. David Fojtík, Ph.D. za odborné vedení a cenné rady při vypracovávání této práce.

## **Anotace**

KOPŘIVA, J. Sledování nosnic pomocí RFID: bakalářská práce. Ostrava: VŠB – Technická univerzita Ostrava, Fakulta strojní, Katedra automatizační techniky a řízení, 2020, 35 s. Vedoucí práce: Fojtík, D.

Bakalářská práce se zabývá návrhem a realizací systému pro sledování nosnic pomocí technologie RFID. Obsahuje rešerši dostupných řešení inteligentního kurníku. Vysvětluje princip fungování RFID, včetně používaných frekvenčních pásem a standardů. Popisuje použité hardwarové prostředky, které byly zvoleny a použity. Dále jsou popsány vývojové programovací prostředky, a to jak použité programovací jazyky, tak i vývojová prostředí, ve kterých bylo pracováno. Hlavní částí je návrh a realizace softwarové části systému sledování nosnic pomocí RFID, které se skládá z webového rozhraní, databáze a algoritmu pro monitorování nosnic. Na konci práce jsou navržena možná rozšíření.

## **Klíčová slova**

RFID, tag, nosnice, sledování, python, PHP, HTML, CSS, SASS, web, Raspberry.

## **Annotation**

KOPŘIVA, J. Monitoring of Laying Hens by RFID: Bachelor Thesis. Ostrava: VŠB –Technical University of Ostrava, Faculty of Mechanical Engineering, Department of Control Systems and Instrumentation, 2020, 35 p. Thesis head: Fojtík, D.

The bachelor's thesis deals with the design and implementation of a system for monitoring laying hens using RFID technology. Contains a search of available intelligent chicken coop solutions. Explains the principle of operation of RFID, including the frequency bands and standards used. Describes the hardware resources used, which have been selected and used. Furthermore, the development programming tools are described, both the used programming languages and the development environments in which it was worked. The main part is the design and implementation of the software part of the laying hen monitoring system using RFID, which consists of a web interface, a database and an algorithm for monitoring laying hens. At the end of the work, possible extensions are suggested.

## **Keywords**

RFID, tag, hen, monitoring, Python, PHP, HTML, CSS, SASS, web, Raspberry.

# Obsah

Úvod .....	1
1 Dostupná řešení inteligentního kurníku pro malochov .....	2
2 Technologie RFID.....	6
2.1 Používaná frekvenční pásma.....	6
2.2 Standardy používané u RFID .....	8
2.3 RFID tag .....	8
2.4 RFID čtečka.....	8
3 Hardwarové prostředky .....	9
3.1 Raspberry Pi 4 Model B.....	9
3.2 RFID čtečka ID12 .....	10
3.3 Použité RFID tagy .....	11
4 Vývojové programovací prostředky .....	12
5 Návrh systému sledování nosnic.....	18
6 Softwarová část.....	21
6.1 Webové rozhraní.....	21
6.2 Databáze .....	25
6.3 Algoritmus monitorovacího systému .....	27
7 Konfigurace a používání systému.....	31
8 Rozšíření a další postup.....	35
9 Závěr.....	36
10 Použitá literatura a zdroje.....	37

## Seznam použitých zkratk

COM	Communication port	Komunikační port
CSS	Cascading Style Sheets	Kaskádové styly
GPIO	General – purpose input/output	Univerzální vstupní/výstupní pin
GUI	Graphical User Interface	Grafické uživatelské rozhraní
HF	High Frequency	Vysoká frekvence
HTML	Hypertext Markup Language	Hypertextový značkovací jazyk
IDE	Integrated Development Environment	Integrované vývojové prostředí
IEEE	Institute of Electrical and Electronics Engineers	
Int	Integer	Celočíselný datový typ
ISO	International Organization for Standardization	Mezinárodní organizace pro standardizaci
LAN	Local Area Network	Lokální počítačová síť
LCD	Liquid Crystal Display	Displej z tekutých krystalů
LED	Light-Emitting Diode	Elektroluminiscenční dioda
LESS	Leaner Style Sheets	Štíhlejší styly
LF	Low Frequency	Nízká frekvence
LPDDR	Low-Power Double Data Rate	Nízkoenergetický dvojnásobný datový tok
PHP	Hypertext Preprocessor	Hypertextový preprocesor
PoE	Power over Ethernet	Napájení přes ethernet
RFID	Radio Frequency Identification	Radiofrekvenční identifikace
SASS	Syntactically Awesome Style Sheets	Syntakticky úžasné styly
SD	Secure Digital	Paměťová karta



SHF	Super High Frequency	Super vysoká frekvence
UHF	Ultra High Frequency	Ultra vysoká frekvence
USB	Universal Serial Bus	Univerzální sériová sběrnice
Varchar	Variable Character Field	Proměnné znakové pole
WAMP	Windows Apache MySQL PHP	Balík vývojových nástrojů
WF	Microwawe	Mikrovlny
Wi-Fi	Wireless Fidelity	Bezdrátová komunikace počítačových sítí

## Úvod

Sledování nosnic pomocí RFID přináší nové možnosti v jejich chovu. Dva hlavní přínosy jsou možnost analýzy snůšky a efektivnější ochrana kurníku.

Analýzou dat, uložených v databázi, lze zjistit, jak která konkrétní nosnice snáší nebo nesnáší, případně zda některá nosnice nezačala kvokat. Tato data jsou získávána RFID čtečkou, umístěnou před místem, kde jsou nosnice zvyklé snášet. Analýzou dat od RFID čteček umístěných před vchodem do kurníku je možné posoudit zhoršenou zdravotní kondici nosnice, zdržuje-li se dlouhodobě uvnitř kurníku. V případě, že se některé nosnice po setmění nevrátí do kurníku systém o tom zobrazí zprávu s varováním.

Pro pochopení toho, jaké jsou v dnešní době dostupné technologie inteligentních kurníků bylo nutné hledat jak mezi tuzemskými, tak i zahraničními výrobci. Po nalezení dostupných řešení byla vytvořena jejich rešerše, kde byla jednotlivá řešení stručně popsána, porovnána a na základě toho vybráno řešení nejlepší.

Před volbou vhodných hardwarových prostředků proběhlo seznámení se samotnou technologií RFID. Na jakém principu funguje, jaká jsou pracovní frekvenční pásma této technologie, které standardy pro ni existují. Bylo zjišťováno, co je to RFID tag a RFID čtečka, k čemu slouží, jaké jsou jejich typy a jak probíhá komunikace RFID tagu s RFID čtečkou. Poté byla zvolena pracovní frekvence a pro ni vhodné RFID tagy a RFID čtečky. Následovala volba platformy pro zpracování dat z RFID čteček a zajišťující chod algoritmu pro monitorování nosnic. V návaznosti na zvolenou platformu byl volen programovací jazyk, pro tvorbu monitorovacího algoritmu, a vhodné vývojové prostředí pro tento jazyk.

Monitorovací systém pro sledování nosnic byl navržen tak, aby bylo možné přistupovat k datům odkudkoliv. Zobrazování dat o pohybu a produkci vajec jednotlivých nosnic je možné pomocí webového rozhraní. S ohledem na to, že velký podíl zobrazovaných webů mají mobilní zařízení, má webové rozhraní responsivní design. To znamená, že se webový obsah přizpůsobuje velikostí i rozložením šířce displeje, na kterém je zobrazován. Vkládání dat do databáze zajišťuje algoritmus běžící na zvolené hardwarové platformě.

## 1 Dostupná řešení inteligentního kurníku pro malochov

Na trhu je dostupných několik různých řešení inteligentního kurníku, která se zaměřují na otevírání a zavírání dvířek. Většina z nich pracuje s časovým ovládáním, ovládáním pomocí čidla detekující světlo nebo jejich kombinací.

### Řešení ChickDoor

Řešení ChickDoor nabízí pouze automatická dvířka ke kurníku. Automatické otevírání a zavírání dvířek je řízeno denním světlem pomocí čidla detekující světlo. Napájení je možno dvěma způsoby a to buď pomocí síťového adaptéru nebo bateriemi AAA. V případě napájení bateriemi je jejich nabití signalizováno tříbarevnou LED diodou. Nežádoucí otevření v nesprávnou dobu je programově ošetřeno, například proti nasvícením měsícem nebo svítlnou a i proti krátkodobému zakrytí světlo detekujícího čidla. V případě překážky v dráze dvířek se dvířka nezavřou. Nabízena jsou buď dvířka s řídicí jednotkou nebo jen řídicí jednotka samostatně. (1)



Obrázek 1.1 Automatická dvířka ChickDoor (1)

### Řešení SI59

Automatický otvírač dveří SI59 zavírá a otevírá dvířka kurníku dvěma způsoby. První způsob funguje na principu detekce denního světla senzorem citlivým na světlo. Druhý způsob využívá k řízení otevírání a zavírání dvířek časovač. Časy otevírání a zavírání lze měnit v menu automatického otvírače. Ovládání a nastavování řídicí jednotky je pomocí 3 tlačítek a LCD displeje, které jsou ukryty uvnitř těla otvírače. Při jiné výšce dvířek, než je výška u

dodávaných, lze v menu výšku změnit a to od 20 cm do 60 cm, aby nedocházelo k nesprávnému zavírání dvířek. Napájení je řešeno pomocí síťového adaptéru. (2)



Obrázek 1.2 Automatický otvírač dveří SI59 (2)

### Řešení *ChickenGuard*

Automatický otvírač dveří *ChickenGuard* otevírá dvířka na stejných principech jako SI59, ale navíc lze způsoby otevírání na časovač a otevírání na detekci světla kombinovat. Ovládá se také pomocí 3 tlačítek a LCD displeje. V tomto případě je ovládání umístěno na vnější straně těla. Napájení je řešeno pomocí baterií AA nebo síťovým adaptérem na 230 V střídavého napětí. (3)



Obrázek 1.3 Řešení *ChickenGuard* (3)

### Řešení Kerbl

Řešení Kerbl nabízí automatické otevírání a zavírání dvířek kurníku řízené buď senzorem citlivým na světlo nebo na základě nastavených časových údajů časovače. U řízení světelným senzorem je nastaven časový úsek, po který musí na senzor svítit světlo, aby nedocházelo k nežádoucímu otevírání dvířek například při krátkodobém nasvícení senzoru svítilnou. Pro

ovládání a nastavování automatického otevírače jsou na jeho přední straně umístěna čtyři tlačítka a jednobarevný LCD displej. Pomocí tlačítek je také možné ručně otevřít nebo zavřít dvířka. Napájení je možné pomocí adaptéru, připojovaného do rozvodné sítě se střídavým napětím 230 V, nebo baterie s napětím 12 V. Maximální výška zdvihu dvířek je 60 cm. Maximální zdvihová hmotnost je 2,5 kg. Výhodou oproti ostatním řešením je možnost rozšíření o ovládání externím signálem přiváděným na přímo k tomu určenou svorkovnici. (4)



Obrázek 1.4 Foto řešení Kerbl v kombinaci s dvířky ChickenGuard, autor Ing. David Fojtík, Ph.D.

## Srovnání

Dostupná řešení inteligentního kurníku byla porovnána na základě vybraných parametrů, kterými jsou zdvihová hmotnost, možnost rozšíření o externí ovládání, možnost řízení časovačem, možnost řízení světelným senzorem, napájecí napětí a cena.

Tabulka 1.1 Porovnání dostupných řešení inteligentního kurníku

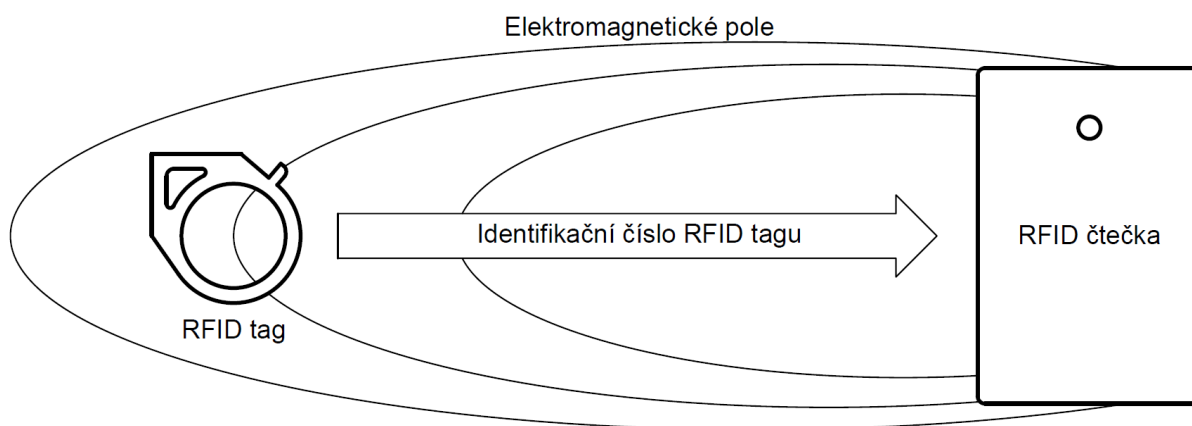
Název	Zdvihová hmotnost [kg]	Možnost rozšíření	Řízení časovačem	Řízení světelným senzorem	Napájecí napětí [V]	Cena [Kč]
ChickDoor	0,5	Ne	Ne	Ano	5	2749
SI59	4	Ne	Ano	Ano	12	4618
ChickenGuard	1	Ne	Ano	Ano	5	4700
Kerbl	2,5	Ano	Ano	Ano	12	4260

Na základě porovnání dostupných řešení inteligentního kurníku byla vybrána jako nejlepší řešení technologie Kerbl. Toto řešení, stejně jako většina ostatních, podporuje řízení časovačem i světelným senzorem, zdvihová hmotnost dvířek, která je zde 2,5 kg, je plně dostačující. Hlavním důvodem výběru byla možnost rozšíření o externí ovládání.



## 2 Technologie RFID

RFID je metoda bezdrátové identifikace založená na komunikaci čtečky s RFID tagem. Čtečka vyše pomocí antény rádiový signál a tag odpovídá svým unikátním identifikačním číslem nebo, pokud se jedná o tag vybavený interní pamětí pro ukládání dat, odesláním obsahu své paměti. Čtečka vysílá v pulzech rádiový signál a když je v dosahu některý tag, pracující na stejné frekvenci jako čtečka, dojde k výměně informací. RFID tagy lze dělit na aktivní a pasivní. Pasivní tagy nemají vlastní zdroj energie a napájeny jsou pouze kondenzátorem, který je nabíjen přeměnou přijímaného rádiového signálu na elektrický proud. V důsledku toho, že jsou pasivní tagy napájeny pouze z kondenzátoru, mají nízký vysílací výkon a musí se proto nacházet v blízkosti čtečky. Aktivní tagy potřebují svůj zdroj energie, ale jejich výhodou je vyšší vysílací výkon. (5)



Obrázek 2.1 Princip RFID

### 2.1 Používaná frekvenční pásma

Technologie RFID pracuje ve čtyřech hlavních frekvenčních pásmech a to v LF, HF, UHF a MW. Jednotlivá frekvenční pásma se liší v rozsazích svých pracovních frekvencí. Signál u každého pásma má rozdílný dosah, jinou přenosovou rychlost a jinou průchodnost překážkami. Obecně platí, čím vyšší frekvence, tím větší dosažitelná vzdálenost přenosu, vyšší přenosová rychlost a nižší schopnost signálu pronikat překážkami. (6) (7)

#### LF pásmo

Pásmo LF je v rozsahu frekvencí od 30 kHz do 300 kHz, kde se nejvíce používají 2 frekvence a to 125 kHz a 134 kHz. Díky nízkým provozním frekvencím je dosahováno malých výkonů a čtecí vzdálenost je nízká, zhruba do 30 cm. V případě použití velmi dlouhých tagů lze

dosáhnout čtecí vzdálenosti až 2 m. V tomto pásmu se vyrábí a používají pouze pasivní tagy, u kterých se paměť nedá přepisovat. Typicky se používá u sledování zvířat a pro identifikaci kusů ve výrobě. (6) (7) (8)

## HF pásmo

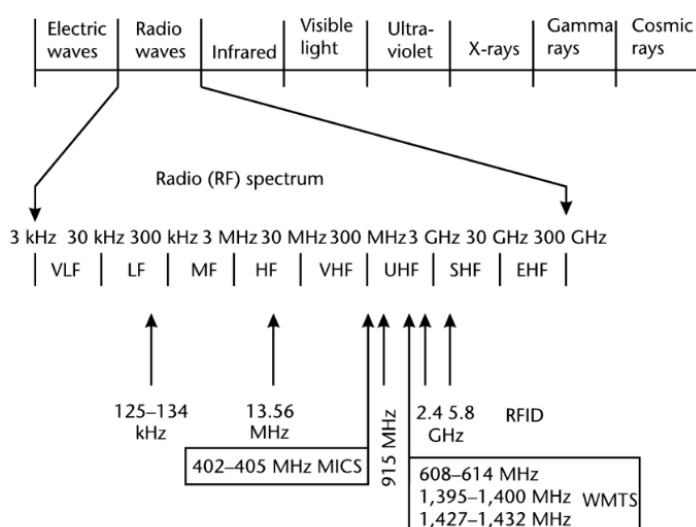
Pásmo HF je v rozsahu frekvencí od 3 MHz do 30 MHz, kde nejvíce používanou frekvencí je 13,56 MHz. Výkon i přenosová rychlost u tohoto pásma je vyšší a proto je možno dosáhnout čtecí vzdálenosti u pasivních tagů až 1 m. Aktivní tagy se v tomto pásmu využívají málo. Typicky se používá u přístupových karet, například v knihovnách nebo prezenčních systémech. (6) (8)

## UHF pásmo

Pásmo UHF je v rozsahu 300 MHz až 3 GHz, nepoužívanější frekvencí tohoto pásma je pro Evropu 868 MHz. Čtecí vzdálenost u pasivních tagů je od zhruba 1 m do 6 m. Při použití aktivních tagů lze dosáhnout čtecí vzdálenosti, za ideálních podmínek, až 30 m. Typicky se používá pro logistické sledování zboží nebo závodníků při závodech. (6) (7) (8)

## MW pásmo

Pásmo MW, také označované jako SHF, je v rozsahu 300 MHz až 300 GHz, používají se v něm 2 frekvence a to 2,45 GHz a 5,8 GHz. U tohoto pásma se používají pouze aktivní tagy kvůli vyšší energetické náročnosti. Čtecí vzdálenost může dosahovat až 100 m. Typicky se používá pro sledování pohybujících se předmětů v reálném čase. (6) (7)



Obrázek 2.2 Rozdělení frekvencí (5)

## **2.2 Standardy používané u RFID**

Pro obecné použití RFID je definován standard ISO 15693. Pro identifikaci osob, zároveň řešící i zabezpečení uložených dat v tagu, je určen standard ISO 1443. Oba tyto standardy jsou platné pro HF pásmo. Pro pásmo UHF je definován standard 18000-6 a pro LF pásmo standard Unique. Standardy ISO – 11784 a ISO – 11785 se zabývají sledováním zvířat. (6) (9)

## **2.3 RFID tag**

RFID tag je nosičem dat a nazývá se také transpondér. Tag se skládá ze 2 hlavních částí, a to z čipu, kde jsou uloženy informace, a antény, která tvoří největší část tagu. U aktivních tagů je součástí integrovaná baterie. Tělo tagu je většinou tvořeno ochranným obalem z PVC nebo skla (skleněná kapsle). Pro některé aplikace se vyrábí RFID tagy zatavené do tenké fólie, opatřené samolepící vrstvou, například v obchodech pro ochranu zboží před odcizením. (6)

## **2.4 RFID čtečka**

RFID čtečka zajišťuje komunikaci mezi RFID tagem a řídicí jednotkou systému. Čtečka se skládá ze 3 částí: antény, rozhraní pro příjem a vysílání signálu a řídicí jednotka čtečky. Pomocí antény je vysílán rádiový signál dané frekvence, který aktivuje RFID tag. U pasivního tagu vyslaný signál z čtečky nabije kondenzátor tagu. Anténa slouží také k příjmu signálu vyslaného tagem. Může být integrovaná přímo v těle čtečky nebo připojená k čtečce externě. Rozhraní pro příjem a vysílání signálu zajišťuje modulaci, demodulaci a přenos rádiového signálu mezi anténou a řídicí jednotkou. Kvůli citlivosti jsou cesty pro vysílání a přijímaný signál odděleny. Řídicí jednotka čtečky zajišťuje zpracování přijatého signálu a komunikaci čtečky s řídicí jednotkou systému. (6)

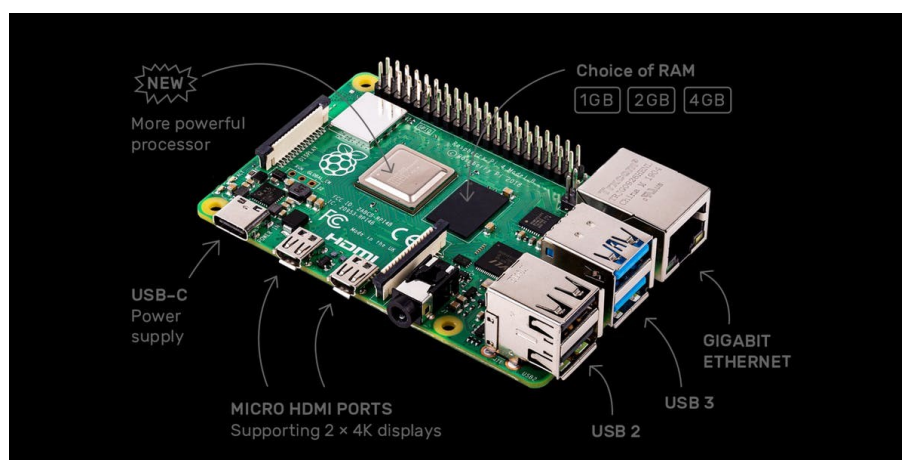
Čtečky jsou stacionární nebo mobilní konstrukce. Stacionární čtečky jsou nepřenosné, je možno k nim připojit více antén, které jsou externí. Stacionární čtečky se využívají například v průmyslové výrobě, skladech nebo ve vchodech obchodů. Mobilní čtečky mají anténu integrovanou v pouzdře čtečky. Jsou malých rozměrů, aby bylo možné je přenášet. Využívat se dají všude, kde je třeba kontrolovat předměty vybavené tagy individuálně. (6)

### 3 Hardwarové prostředky

Pro sledování nosnic byl použit jednodeskový počítač Raspberry Pi 4 Model B, RFID čtečky ID-12 a tagy ve tvaru kroužku.

#### 3.1 Raspberry Pi 4 Model B

Jako hardwarová platforma byl zvolen jednodeskový počítač Raspberry Pi 4 Model B. Oproti předešlému modelu Raspberry Pi 3 Model A+ je mnohem výkonnější, nabízí lepší konektivitu a lze jej použít jako stolní počítač. Operační systém běžící na této platformě se nazývá Raspbian.



Obrázek 3.1 Raspberry Pi 4 Model B

Zdroj: <https://www.inverse.com/article/57196-raspberry-pi-4>

#### Technické parametry

Počítač má velice dobrou konektivitu. Pro komunikaci s periferiemi, kterými u sledování nosnic jsou RFID čtečky, jsou k dispozici 4 USB porty typu A, a to 2 USB 3.0 a 2 USB 2.0. Pro bezdrátovou komunikaci s počítačem je možné použít Bluetooth 5.0 nebo Wi-Fi (IEEE 802.11b/g/n/ac) na frekvencích 2,4 GHz a 5 GHz. Pro komunikaci kabelově po síti je k dispozici Gigabit Ethernet s konektorem RJ45. Pomocí dvou portů Micro HDMI lze připojit dva monitory, jeden může zobrazovat v kvalitě 4K. Provozní teplota počítače může být od 0°C do 50°C, proto je nutné ho ochránit před mrazem. (10)

Napájecí napětí počítače je 5 V, stejnosměrné. Napájet jej lze přes port USB typu C, přímo přes piny 40 kolíkového GPIO headeru nebo v případě připojeného síťového kabelu pomocí PoE. (10) (11)

Operační paměť počítače má velikost 4 GB a je typu LPDDR4. Pro načítání operačního systému a ukládání dat se používá externí Micro SD karta, pro kterou má počítač integrovaný slot. Minimální doporučená velikost karty je 8GB a karta by měla být třídy 4 (rychlost zápisu 4 MB/s) a vyšší. (10)

### 3.2 RFID čtečka ID12

Čtečka ID12-USB-SE-BOX pracuje ve frekvenčním pásmu LF, na frekvenci 125 kHz a řídí se standardem Unique. Připojuje se pomocí USB kabelu, který je dlouhý 1,8 m. Čtečka se při komunikaci nechová jako USB zařízení, ale jako zařízení pracující na sériové lince COM. Tělo čtečky má tvar kvádru a rozměrech 50 x 35 x 20 mm. Na čelní straně čtečky je umístěna červená LED dioda, která funguje jako indikátor přijetí signálu od RFID tagu. Čtecí vzdálenost udávaná výrobcem by měla být až 12 cm, ale při testování s použitými RFID tagy byla čtecí vzdálenost pouze 3,5 cm. (9)



*Obrázek 3.2 Použitá RFID čtečka*

### 3.3 Použité RFID tagy

Pro použití na sledování nosnic byly vybrány tagy ve tvaru kroužku pro nasazení na nohu nosnice. Tagy mají vnitřní průměr 19 mm a výšku 11 mm. Tělo tagu je vyrobeno z ABS plastu. Provozní teploty tagů se mohou pohybovat od  $-45^{\circ}\text{C}$  do  $80^{\circ}\text{C}$ , což je pro toto použití plně dostačující. Pracovní frekvence tagů je 125 kHz. Vzdálenost, na kterou lze tagy číst je udávána výrobcem od 2 cm do 20 cm. Při prvotním testování se čtečkou ID12 byla čtecí vzdálenost 3,5 cm. Vnitřní paměť tagů je 64 bitů. Tagy splňují standardy ISO – 11784 a ISO – 11785.



*Obrázek 3.3 Použitý RFID tag*



*Obrázek 3.4 Nosnice s nasazeným RFID tagem*



## 4 Vývojové programovací prostředky

Pro vytvoření programu, zaznamenávajícího pohyby nosnic a jejich ukládání do databáze na zvolené hardwarové platformě, kterou je Raspberry Pi, byl zvolen programovací jazyk Python. Webové rozhraní bylo vytvářeno pomocí HTML a PHP. Vizuální styl webového rozhraní byl upravován pomocí CSS.

### Procedurální a objektové programování

Procedurální styl programování spočívá ve psaní kódu, který je nedělený. Pro opakovatelnost kódu, tak aby nebyl v programu duálně, se využívají funkce. Procedurální programování se hodí spíše pro menší aplikace (zhruba do 500 řádků kódu), u rozsáhlejších programů se kód stává méně přehledný a neefektivní. (12)

V objektově orientovaném programování se kód sdružuje do objektů. Objekty mají svoje atributy a metody. Atributy popisují vlastnosti objektu, metody popisují činnosti, které může objekt provést. Objekty mohou být volané jako instance tříd. Třídy jsou konstruktory objektů, pomocí nichž se objekty volají. (12)

### Python

Programovací jazyk Python patří mezi jeden z nejjednodušších na naučení. Programy psané v Pythonu jsou přehledné a snadné na čtení a orientaci v nich. Soubor s programem psaným v Pythonu má příponu „.py“. Python je hodně expresivním jazykem, takže v mnoha případech kód napsaný v Pythonu je kratší a méně složitý, než kód se stejnou funkcionalitou napsaný v jazyku C++ nebo Java. Nejedná se o jazyk, který by byl přímo vázaný na jedinou platformu, na rozdíl například od jazyku C#, proto může být stejný program napsaný v Pythonu spuštěn na mnoha platformách, bez nutnosti jakýchkoliv nástaveb a kompilace. Funkcionality, které jsou specifické jen pro některé platformy jsou často zahrnuty ve standardní knihovně Pythonu. Jádro jazyku Python je objektově orientované, ale programy lze psát jak objektově orientované, tak procedurální. Struktura psaného programu se vytváří pomocí velikosti odsazení. (12)

### Knihovny pro Python

Již standardní knihovna Pythonu je velmi rozsáhlá – dovoluje pomocí krátkého programu o pár řádcích například stahovat soubory z internetu nebo práci s archivovanými daty. Pokud

pro dané řešení standardní knihovna není dostačující, jsou pro její rozšíření k dispozici tisíce knihoven třetích stran. (12)

## PHP

Programovací jazyk PHP je skriptovací jazyk nejčastěji používaný pro komunikaci s databází a přístup k jejím datům. Jazyk PHP je původně založený na jazyce C. Soubor s programem psaným v PHP má příponu „.php“. Program psaný v PHP se provádí na straně serveru, generuje HTML, které je odesíláno směrem ke klientovi. Stejně jako Python je i PHP jednoduchým jazykem pro začínající programátory. Při vyvíjení na lokální stanici je nutné na této stanici vytvořit virtuální lokální server. Při vývoji webového rozhraní pro sledování nosnic byl virtuální server vytvořen pomocí programu WampServer. Každá proměnná je identifikována tak, že její název začíná znakem dolar – „ \$ “. Všechny řádky, na kterých se nachází nějaký kód programu, musí být zakončeny znakem středník – „ ; “, pokud tak není učiněno je vypsaná chybová hláška a program se dále od tohoto místa neprovede. (13)

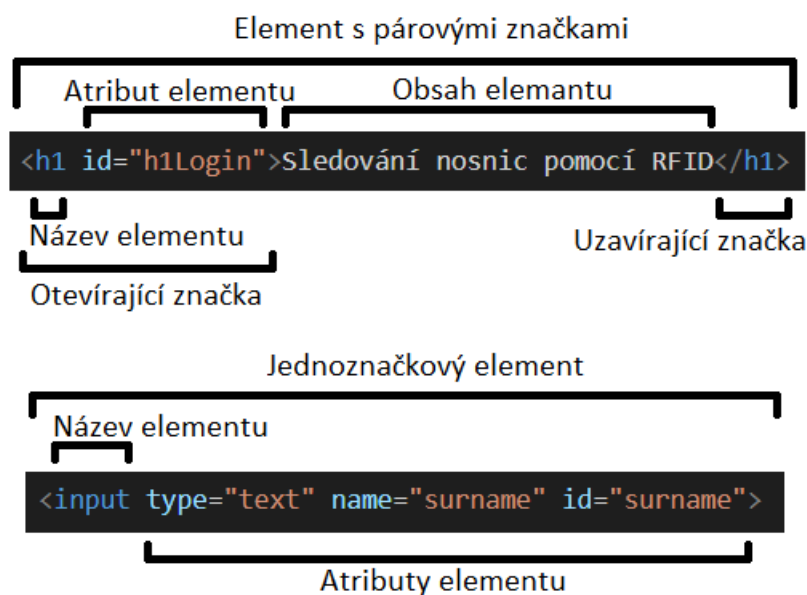
```
$userId = 1;
$reqUser = "SELECT `username`,`name`,`surname`,`phone` FROM `user` WHERE `id_user`= ?";
$stmtUser = mysqli_prepare($connect,$reqUser);
mysqli_stmt_bind_param($stmtUser,'i',$userId);
mysqli_stmt_execute($stmtUser);
mysqli_stmt_store_result($stmtUser);
mysqli_stmt_bind_result($stmtUser,$userName,$name,$surname,$phone);
mysqli_stmt_fetch($stmtUser);
mysqli_stmt_close($stmtUser);
mysqli_close($connect);
```

Obrázek 4.1 Ukázka části PHP programu, konkrétně výběr údajů uživatele pomocí předpřipravených výrazů

## HTML

Jedná se o značkovací jazyk, pomocí kterého se vytváří struktura webových stránek. Díky HTML má webový prohlížeč informace o struktuře stránky a je tak zajištěno její správné formátování a zobrazení. Soubor s HTML kódem má příponu „.html“. Stránka napsaná pomocí HTML je tvořena z do sebe vnořovaných elementů. Elementy lze dělit na dvě hlavní skupiny, a to na elementy s párovými značkami a na jednoznačkové elementy. Elementy s párovými značkami mají vždy otevírací značku a uzavírací značku. Otevírací značka začíná symbolem menší než – „<“, po ní následuje název elementu, atributy elementu, pokud daný element nějaké má a zakončena je znakem větší než – „>“. Uzavírací značka začíná, stejně jako otevírací, znakem menší než, za nímž se nachází znak lomeno – „/“, dále následuje název

elementu a ukončující znak větší než. Jednoznačkové elementy mají pouze jednu značku, která odpovídá popisu otevírající značky elementu s párovými značkami. (14)



Obrázek 4.2 Ukázka HTML elementů s popisem jejich částí

## CSS a SASS

Kaskádové styly jsou jazykem, pomocí kterého lze upravovat u HTML elementů jejich vzhled a pozici v rámci stránky. Existují tři možnosti, jak CSS kód zapisovat. Prvním, nejprimitivnějším a nejméně používaným, způsobem je vepisování CSS kódu přímo do těla HTML elementu do atributu „style“, který se nazývá inline. Druhý způsob zápisu je interní. V tomto případě je CSS kód napsán jako obsah elementu s párovými značkami s názvem „style“, který je umístěn v hlavičce HTML dokumentu. Třetím způsobem je externí zápis. Kód CSS je uložený v samostatném souboru s příponou „.css“. Aby se externí stylování na dané stránce projevilo, musí být v její hlavičce element „link“ odkazující na externí CSS soubor. (15)

### Inline zápis

```
<h1 style='color:green;'>Nosnice</h1>
```

### Interní zápis

```
<style>
  h1{
    color:green;
  }
</style>
```

### Odkaz souboru externího zápisu

```
<link rel="stylesheet" type="text/css" href="./css_rfid_hens/main.css">
```

Obrázek 4.3 Ukázky zápisů CSS

## Thonny IDE

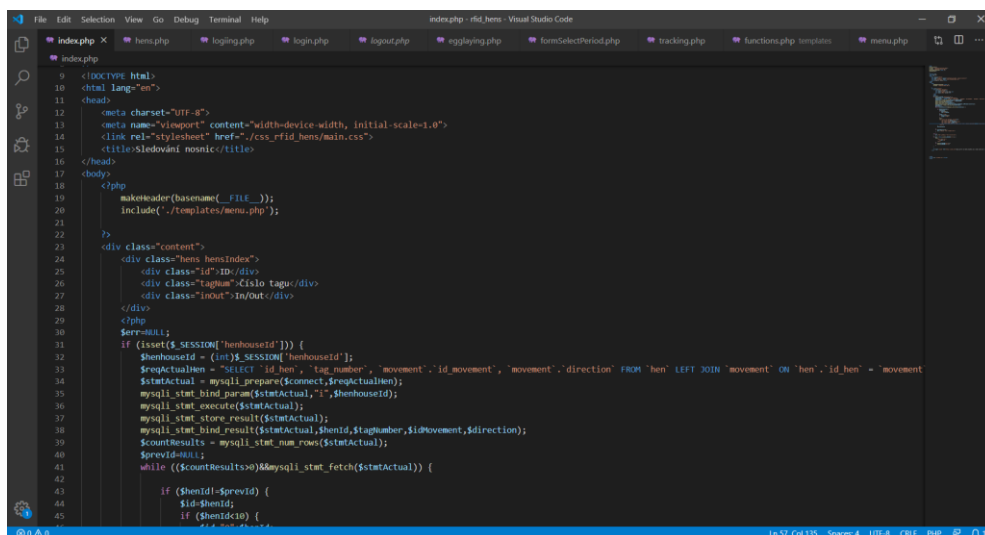
Okno programu je rozděleno na pět sekcí. V horní části okna se nachází panel základní nabídky společně s ikonami rychlých výběrů, jako je například vytvoření nového souboru, spuštění kódu nebo zastavení kódu. Na pravé straně okna jsou pod sebou sekce s proměnnými a jejich hodnotami a sekce s nápovědou. Nalevo od těchto dvou sekcí je oblast pro psaní kódu, která je největší a zabírá zhruba polovinu plochy celého okna, pod kterou se nachází výstupní konzole.



## Visual Studio Code

Program Visual Studio Code od společnosti Microsoft je editor zdrojového kódu a nejedná se o IDE. Dostupný je pro operační systémy Windows, Linux i macOS. V základu je vestavěná podpora jazyků JavaScript, TypeScript a Node.js. Dostupných je mnoho rozšíření pro další jazyky jako je například PHP nebo Python. (18)

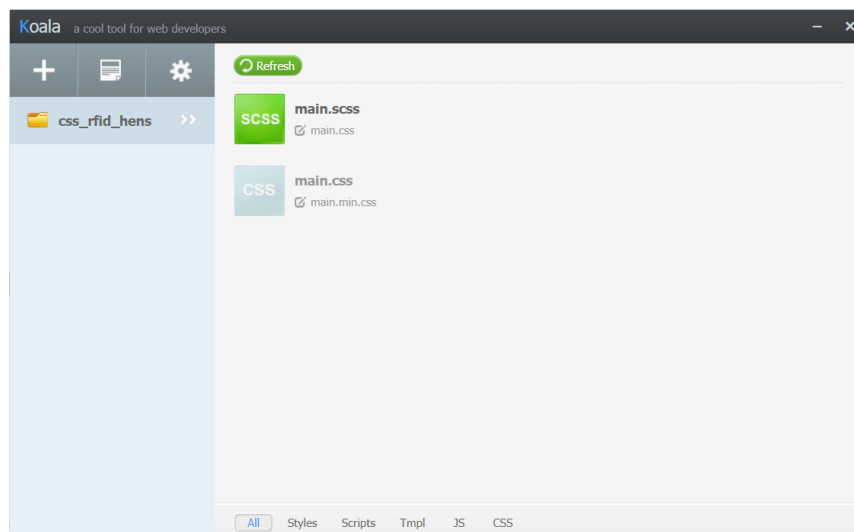
V základním rozložení programu největší část okna zabírá prostor pro editování zdrojového kódu, který se rozprostírá od středu okna směrem do stran. Nalevo od tohoto prostoru se nachází boční panel pro zobrazování podoken, jako je například podokno se všemi soubory aktivního adresáře, podokno vyhledávání v kódu souborů aktivního adresáře nebo podokno pro správu a instalaci rozšíření. Nad prostorem pro editaci kódu je lišta s aktuálně otevřenými soubory. V horní části okna se nachází panel hlavní nabídky programu sloužící pro jeho ovládání.



Obrázek 4.5 Okno programu Visual Studio Code

## Koala

Program Koala je volně dostupným kompilátorem, u kterého veškeré ovládání probíhá přes jeho GUI. Dostupný je pro operační systémy Windows, Linux a macOS. Kompilovat lze soubory s kódem v jazycích LESS, SASS, Compass a CoffeeScript. Koala kompiluje v reálném čase, konkrétně čeká na nové změny v souboru a po jejich zjištění automaticky kompiluje.



## WampServer

WampServer je program umožňující vytváření virtuálního serveru na lokálním počítači. Je určen pro operační systém Windows. Vytvoření lokálního webového serveru dovoluje vyvíjet a testovat webové rozhraní ještě před tím, než je nahráno na reálný server připojený k internetu. Tento program se skládá ze třech částí, kterými jsou Apache – spouští webový server, MySQL – vysokorychlostní databáze a PHP.

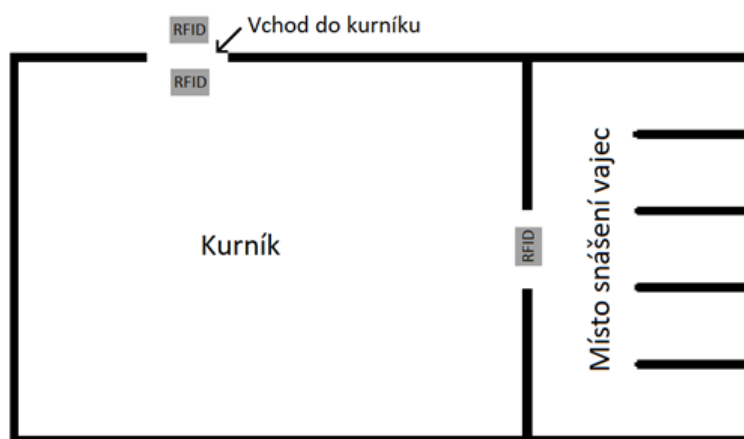


## 5 Návrh systému sledování nosnic

Žádné z dostupných řešení inteligentního kurníku neřeší při zavírání dvířek přítomnost či nepřítomnost nosnic v kurníku. Může nastat případ, kdy se některá z nosnic nestihne přemístit před uzavřením dvířek do vnitřní části kurníku a bude tak přes noc vystavena nepřízní počasí a nebezpečí napadení predátorem. Sledování nosnic přináší možnost analýzy kondice nosnic, kvočí-li některá nosnice a zjištění, jak která nosnice snáší, zde by ale bylo vhodné ověřovat, zda nosnice skutečně snesla vejce, nebo se pouze zdržovala v místě snášení.

Možný zhoršený zdravotní stav nosnice lze odhadovat podle častého zdržování se uvnitř kurníku v porovnání s ostatními nosnicemi nebo výrazně vyššího procenta času stráveného uvnitř oproti procentu času stráveného venku. Rozpoznání toho, že nosnice kvočí je možné posuzovat podle vysokého času stráveného v místě snášení vajec, aniž by nosnice toto místo opouštěla a tím zvyšovala počet snášení.

Pro sledování nosnic byla zvolena technologie RFID. Aby bylo zajištěno účinné sledování polohy jednotlivých nosnic, zda jsou uvnitř nebo vně kurníku, jsou použity 2 RFID čtečky. Čtečky jsou umístěny ve vchodu do kurníku tak, aby nosnice prošla nejprve přes jednu a pak přes druhou. Z rozdílných časů průchodů přes čtečky u vchodu lze určit směr pohybu nosnice a určit tak její polohu. Pro vyhodnocování snášení vajec je umístěna RFID čtečka před místem snášení vajec. Pro spolehlivé určení, zda nosnice vejce skutečně snesla by bylo nutné kontrolovat fyzickou přítomnost vejce, například použitím váhového senzoru.



Obrázek 5.1 Návrh rozmístění čteček

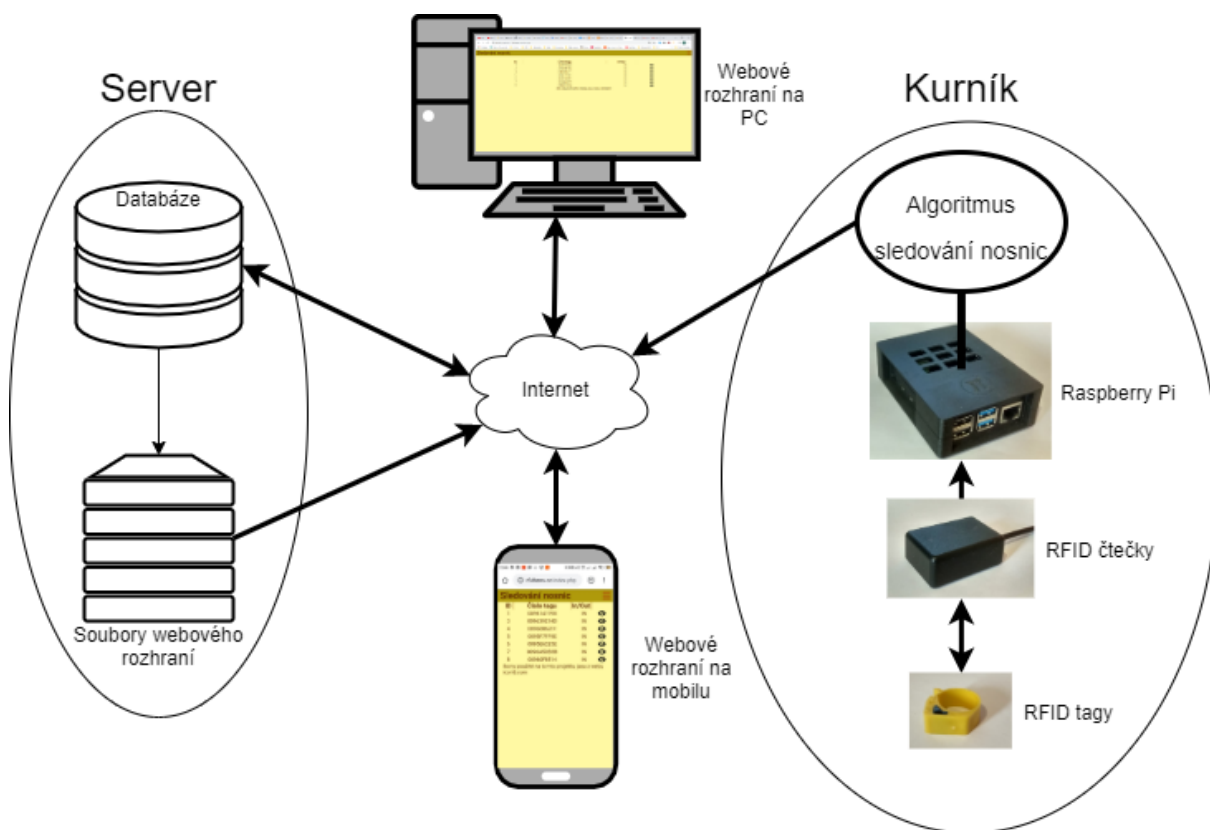
Nejvhodnější umístění RFID čteček je na podlaze průchodů, vzhledem k nízké čtecí vzdálenosti. V případě použití výkonnějších čteček by bylo možné je umístit na stěnu

průchodu. V rámci vchodu do kurníku i vchodu do místa snášení je nutné čtečky umístit tak, aby docházelo ke spolehlivému čtení tagů.

Umístění platformy Raspberry v rámci kurníku je závislé na vzájemné poloze vchodu do kurníku a vchodu do místa snášení. "

## Rozložení systému

Systém je koncipován tak, aby bylo možné přistupovat k datům odkudkoliv. Z toho důvodu jsou soubory webového rozhraní a databáze uloženy na webovém serveru. Platforma Raspberry Pi, RFID čtečky a nosnice s RFID tagy se nachází v kurníku. Přenášení dat z kurníku, o pohybu a snášení nosnic, probíhá přes internet a je zajišťováno algoritmem pro sledování nosnic běžícím na platformě Raspberry Pi. Pro zobrazení dat o pohybu nosnic nebo o snášení vajec se stačí připojit pomocí webového prohlížeče k webovému rozhraní.








Obrázek 5.2 Systémový pohled

## Grafický návrh

Grafický návrh rozložení webového rozhraní byl vytvořen nejprve pro mobilní zařízení, od kterého se později odvíjelo rozložení pro zařízení s vyšší šířkou obrazovky. Na každé stránce se

nachází v horní části lišty s názvem dané stránky. Na pravé straně této lišty je umístěno tlačítko pro zobrazení menu. Pod lištou je prostor pro vypisování obsahu dané stránky.

Sledování nosnic			
id	Tag number	Stav	
1	Tag number	in	
2	Tag number	out	
3	Tag number	in	
...	...	...	

Obrázek 5.3 Grafický návrh základního rozvržení stránky Home

## 6 Softwarová část

Softwarová část se skládá ze tří částí, a to z algoritmu pro evidenci nosnic, webového rozhraní a databáze. Program s algoritmem pro evidenci nosnic je uložen a spouštěn na platformě Raspberry Pi. Webové rozhraní a databáze jsou uloženy na serveru.

### 6.1 Webové rozhraní

Webové rozhraní je tvořeno šesti hlavními a několika vedlejšími stránkami. Všechny hlavní stránky jsou přímo přístupné z menu, které se otevírá kliknutím na ikonu v pravém horním rohu. Mezi hlavní stránky patří „Home“, „Pohyb nosnic“, „Snůška“, „Kurníky“, „Nosnice“ a „Profil“. Mezi vedlejší patří stránky pro přihlášení a odhlášení uživatele, zobrazení detailů nosnice a kurníku, úpravu nosnice, kurníku a profilu. Mezi stránkami se přepíná pomocí menu, které se zobrazí po kliknutí na ikonu v pravém horním rohu každé stránky. Všechny stránky, s výjimkou stránky pro přihlášení, jsou dostupné až po přihlášení uživatele. Pro nové uživatele je k dispozici registrace. Nový uživatel má možnost již při registraci zadat unikátní klíč kurníku pro jeho přiřazení k nově vytvářenému účtu. Pokud není k uživatelskému účtu přiřazen žádný kurník, v žádné ze sekcí se nevypisují data. Každý účet může mít přiřazeno více kurníků a zároveň každý kurník může být přiřazen k více účtům.



Obrázek 6.1 Menu

#### Home

Stránka Home se zobrazí automaticky po přihlášení. Na této stránce se vypisují všechny nosnice uložené v databázi přiřazené k aktivnímu kurníku, u kterých byl již zaznamenán pohyb. Konkrétně se zde vypisuje ke každé nosnici její „id“ – které značí pořadí zápisu nosnice do

databáze, „číslo tagu“ – hexadecimální zápis identifikačního čísla RFID tagu a „in/out“ – vyjadřující, zda se daná nosnice aktuálně nachází uvnitř nebo vně kurníku. Hlavním účelem této stránky je rychlý přehled aktuálního stavu všech nosnic aktivního kurníku. Pokud není k danému uživateli přiřazen žádný kurník, vypíše se pouze chybová hláška.



ID	Číslo tagu	In/Out	
1	0096141998	IN	👁
3	009639E14D	IN	👁
4	00960862FF	IN	👁
5	0095F7FF9E	OUT	👁
6	0095E62E5E	IN	👁
7	0096A50B3B	IN	👁
8	00960F8E14	IN	👁

Ikony použité na tomto projektu jsou z webu icon8.com

Obrázek 6.2 Náhled na stránku Home

## Pohyb nosnic

Na stránce Pohyb nosnic jsou ke každé nosnici z aktivního kurníku vypisována data o jejím pohybu, pokud byl nějaký pohyb za vybrané období zaznamenán. U každé nosnice se vypisuje její „id“, „in“ – čas strávený uvnitř kurníku v hodinách a minutách a počet procent strávených uvnitř vypočítaný z celkového počtu hodin z vybraného období a „out“ – čas strávený vně kurníku v hodinách, minutách a procentech.

Čas strávený uvnitř kurníku za vybrané období se vypočítává pro každou nosnici zvlášť a to tak, že se k času uvnitř přičte časový rozdíl mezi pohybem směrem dovnitř a pohybem směrem ven z kurníku. Pro přičtení rozdílu je nutné splnit podmínku stejného datumu obou pohybů. Čas strávený vně kurníku se vypočítává stejným způsobem, kde se přičítá časový rozdíl mezi pohybem směrem ven a pohybem směrem dovnitř.

K nepřesnostem zde může docházet nezaznamenání jednoho z pohybů, ať už směrem dovnitř nebo ven z kurníku, do databáze v důsledku chybného přečtení jednou z RFID čteček. Pokud jsou u dané nosnice zaznamenány po sobě jdoucí dva pohyby stejného druhu, například dvakrát vstup do kurníku, není způsob, jak určit kde mezi těmito pohyby byl pohyb opačného směru a tento čas se nezapočítává.

Výběr období, pro které se vypisuje pohyb nosnic, je pomocí elementu select. Lze vybírat z možností „Dnes“, „Týden“ – vypisuje aktivitu za posledních 7 dní, „Měsíc“ – vypisuje aktivitu v aktuálním měsíci, „Datum“ – vypisuje aktivitu za vybraný den, „Období“ – vypisuje aktivitu v rozmezí 2 vybraných datumů a „Vše“ kde vypíše veškerou aktivitu v databázi.



ID	IN	OUT
1	1 hod 50 min 71.3%	0 hod 44 min 28.7%
3	1 hod 49 min 86.18%	0 hod 18 min 13.82%
4	1 hod 48 min 82.3%	0 hod 23 min 17.7%
5	1 hod 49 min 82.76%	0 hod 23 min 17.24%
6	1 hod 31 min 60.71%	0 hod 59 min 39.29%
7	1 hod 56 min 93.21%	0 hod 8 min 6.79%
8	1 hod 53 min 88.58%	0 hod 15 min 11.42%

Obrázek 6.3 Náhled na stránku Pohyb nosnic


## Snůška

Na stránce Snůška jsou vypisovány ke každé nosnici, z aktivního kurníku, „id“, „Počet“ – počet výskytů v místě snášení, symbolizující počet snášení a „Čas“ – čas strávený v místě snášení, pokud byl zaznamenán nějaký výskyt v místě snášení za vybrané období. Data vypisovaná na této stránce jsou pouze informativního rázu, protože nedochází k ověřování, zda nosnice skutečně snesla vejce při pobytu uvnitř místa snášení.

Způsob výpočtu času snášení je podobný jako u Pohybu nosnic. K celkovému času snášení se přičítá rozdíl času výstupu a času vstupu do místa snášení. Při každém přičtení času se zároveň zvýší počet výskytů v místě snášení o jedna. Zde k nepřesnostem vypisovaných dat času a počtu nedochází, protože pohyby v místě snášení, dovnitř i ven, jsou zapisovány společně. Pokud dojde k chybě při čtení RFID čtečkou, data se do databáze nezapíší.



Výběr období, pro které se vypisuje aktivita v místě snášení probíhá stejným způsobem jako na stránce Pohyb nosnic.

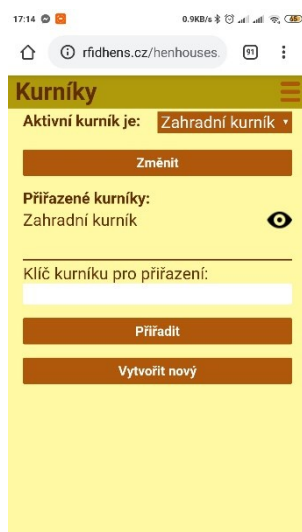


ID	Počet	Čas
1	2	0 hod 22.1 min
3	1	0 hod 7.9 min
4	1	0 hod 11.7 min
5	1	0 hod 3.3 min
6	1	0 hod 18.3 min
7	1	0 hod 12.2 min
8	1	0 hod 17.3 min

Obrázek 6.4 Náhled na stránku Snůška

## Kurníky

Stránka kurníky slouží pro správu kurníků přiřazených k profilu. V horní části stránky se nachází formulář pro změnu aktivního kurníku. Tento formulář se skládá pouze z elementu „select“, kde lze vybírat z kurníků přiřazených k profilu a z potvrzovacího tlačítka „Změnit“. Pod formulářem pro aktivní kurník jsou vypisovány všechny přiřazené kurníky. Pod formulářem se nachází tlačítko odkazující na stránku pro vytvoření kurníku nového. Tato sekce se zobrazuje jako jediná, pokud k účtu není přiřazen žádný kurník.



Aktivní kurník je: Zahradní kurník

Změnit

Přiřazené kurníky:  
Zahradní kurník

Klíč kurníku pro přiřazení:

Přidat

Vytvořit nový

Obrázek 6.5 Náhled na stránku Kurníky

## Nosnice

Na stránce nosnice je vypisován přehled všech nosnic, které náležejí kurníkům připojeným k uživatelskému účtu. U každé zde vypisované nosnice je uvedeno její ID, číslo RFID tagu a kurník, pod který nosnice spadá.



ID	Číslo tagu	Kurník
1	0096141998	Zahradní kurník
3	009639E14D	Zahradní kurník
4	00960862FF	Zahradní kurník
5	0095F7FF9E	Zahradní kurník
6	0095E62E5E	Zahradní kurník
7	0096A50B3B	Zahradní kurník
8	00960F8E14	Zahradní kurník

Vytvořit novou

Obrázek 6.6 Náhled na stránku Nosnice

## 6.2 Databáze

Databáze slouží k ukládání informací o uživateli, nosnicích, kurnících a k zaznamenávání pohybů nosnic. Jedná se o relační typ databáze a skládá se z šesti tabulek – hen, henhouse, movement, typemovement, user a user\_to\_henhouse.

### Hen

Tabulka Hen slouží k evidování jednotlivých nosnic a jejich parametrů. Tabulka má šest atributů. Atributy jsou „id\_hen“ – identifikační číslo nosnice, je typu int, primárním klíčem tabulky a při zadávání nové nosnice se nevyplňuje protože je AUTO\_INCREMENT (vyplňuje se automaticky a při každém novém záznamu se zvětší o jedna), „tag\_number“ – hexadecimální zápis čísla RFID tagu, je typu varchar a je unikátním klíčem, „breed“ – plemeno nosnice, je typu varchar, „hatched“ – datum vylíhnutí nosnice, je typu date, „description“ – bližší popis nosnice, je typu text a „henhouse\_id“ – „id“ kurníku, kterému nosnice náleží, je typu int a je cizím klíčem k tabulce henhouse.

## Henhouse

V tabulce henhouse jsou ukládány jednotlivé kurníky. Tabulka má tři atributy a to „id\_henhouse“ – identifikační číslo kurníku, je typu int, primárním klíčem tabulky a je AUTO\_INCREMENT, „name“ – název kurníku, je typu varchar a je unikátním klíčem a „place“ – místo kde se kurník nachází, je typu varchar.

## Movement

Tabulky movement má pět atributů a jsou v ní ukládány všechny pohyby nosnic vzhledem ke kurníku a místu snášení. Atributy jsou „id\_movement“ – identifikační číslo pohybu, je typu int, primárním klíčem tabulky a je AUTO\_INCREMENT, „hen\_id“ – „id“ nosnice, které je daný pohyb, je typu int a je cizím klíčem k tabulce hen, „typemovement\_id“ – „id“ typu pohybu, je typu int a je cizím klíčem k tabulce typemovement, „direction“ – směr pohybu nosnice (0 = směr ven, 1 = směr dovnitř), je typu boolean a „datetime“ – datum a čas pohybu, je typu datetime.

## Typemovement

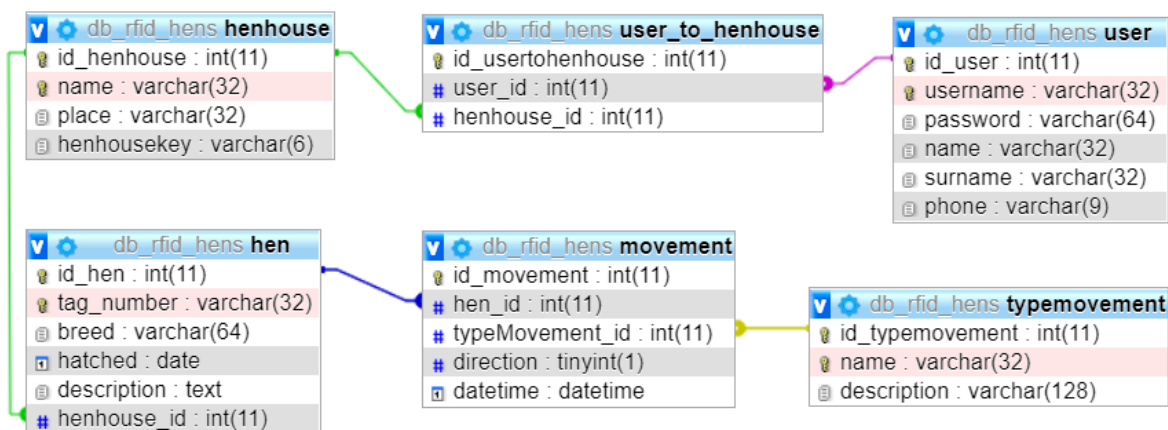
V tabulce typemovement jsou typy pohybů. Tabulka má tři atributy a to a to „id\_typemovement“ – identifikační číslo typu pohybu, je typu int, primárním klíčem tabulky a je AUTO\_INCREMENT, „name“ – název typu pohybu, je typu varchar a je unikátním klíčem a „description“ – popis typu pohybu, je typu text.

## User

V tabulce user jsou ukládány informace k jednotlivým uživatelům, tabulka má šest atributů. Atributy jsou „id\_user“ – identifikační číslo uživatele, je typu int, primárním klíčem tabulky a je AUTO\_INCREMENT, „username“ – uživatelské jméno uživatele, je typu varchar a je unikátním klíčem, „password“ – heslo uživatele zašifrované pomocí SHA1, je typu varchar, „name“ – jméno uživatele, je typu varchar, „surname“ – příjmení uživatele, je typu varchar a „phone“ – telefonní číslo uživatele, je typu varchar.

## User\_to\_henhouse

Tabulka user\_to\_henhouse slouží pouze pro propojení  $n:m$  tabulek user a henhouse, to znamená pro připojení více uživatelům k více kurníkům.



Obrázek 6.7 Schéma databáze

### 6.3 Algoritmus monitorovacího systému

Algoritmus běží na platformě Raspberry Pi a je navržen pro tři RFID čtečky, dvě z nich jsou pro vchod kurníku, třetí pro průchod k místu snášení. Celý algoritmus je psaný procedurálním stylem v jazyce Python a je dělen do funkcí.

#### Funkce readFromCom

Hlavní funkce s názvem „readFromCom“ je volána v nekonečné smyčce v rámci vlákna vytvořeného pomocí knihovny „threading“. Tato funkce má čtyři parametry: „doors\_in“ – seriový port ke kterému je přiřazená vnitřní vchodová čtečka, „doors\_out“ – seriový port vnější vchodové čtečky, „doors\_laying“ – seriový port čtečky místa snášení a „sleepTime“ – čas ve vteřinách, na který se funkce uspává.

Na začátku každého zavolání funkce „readFromCom“ se zjišťuje, zda na některém ze sériových portů nejsou ve vyrovnávací paměti načtená data, pokud ne, funkce je uspána na čas definovaný parametrem „sleepTime“ a po uplynutí tohoto času je v rámci vlákna funkce zavolána znovu. Pokud jsou ve vyrovnávací paměti data, tak jsou přečtena, uložena do příslušných proměnných a volají se funkce „checkMovement“ a „henLaying“. Funkce „checkMovement“ se volá dvakrát po sobě, jednou pro pohyb dovnitř kurníku a podruhé pro pohyb ven z kurníku. Následně se všechny vyrovnávací paměti resetují.

#### Funkce checkMovement

Funkce „checkMovement“ slouží k určení, zda došlo k pohybu nosnice nebo ne. Parametry funkce „checkMovement“ jsou „readData“ – data z čtečky pro příslušný směr pohybu, ze

kterých se separuje číslo tagu, „readTime“ – čas čtení, „arrayInOut“ – dvourozměrné pole, obsahující čísla tagů a časy průchodů, opačného směru pohybu a „direction“ – směr pohybu. Tato funkce může být volaná pro pohyb nosnice dovnitř kurníku i ven. Směr pohybu je rozlišován parametrem „direction“.

Na začátku funkce je kontrolován parametr „readData“. Pokud je nenulový, funkce pokračuje, jinak končí. Následuje volání funkce „separateTagNum“ s parametrem „readData“, jejíž výstup je uložen do proměnné „tagNum“ představující číslo přečteného RFID tagu. Po zjištění čísla tagu se prohledává dvourozměrné pole opačného směru, než pro který je funkce volaná, a hledá se shoda hodnoty proměnné „tagNum“ s některým z tagů uložených v prvním rozměru pole. V případě nalezení shody je přečten z příslušné pozice pole jeho druhý rozměr, kterým je čas čtení v době ukládání do pole. Tento čas je odečten od času uloženého v parametru „readTime“ a rozdíl je porovnán s hodnotou globální proměnné „maxMovementTime“. Pokud je rozdíl menší nebo roven, číslo tagu uložené v poli je smazáno a zavolá se funkce „movementToDB“ s parametry „tagNum“, „readTime“ a „direction“.

### **Funkce henLaying**

Funkce „henLaying“ je určena ke kontrole, zda došlo ke snášení. Funguje podobným způsobem jako funkce „checkMovement“. Funkce má 3 parametry a to „readData“ – data z čtečky místa snášení, ze kterých se separuje číslo tagu, „readTime“ – čas čtení, „arrayLaying“ – dvourozměrné pole, obsahující čísla tagů a časy průchodů nosnic do místa snášení.

Na začátku funkce je, stejně jako u funkce „checkMovement“, kontrolován parametr „readData“, je volána funkce „separateTagNum“, s výstupem do proměnné „tagNum“, s parametrem „readData“. Poté se prohledává dvourozměrné pole určené pro snášení, hledá se shoda hodnoty proměnné „tagNum“ s některým z tagů uložených v prvním rozměru pole. V případě nalezení shody je přečten z příslušné pozice druhého rozměru pole čas, který je odečten od času z parametru „readTime“. Pokud je rozdíl větší nebo roven hodnotě globální proměnné „minLayingTime“, číslo tagu uložené v poli je smazáno a zavolá se funkce „movementToDB“.

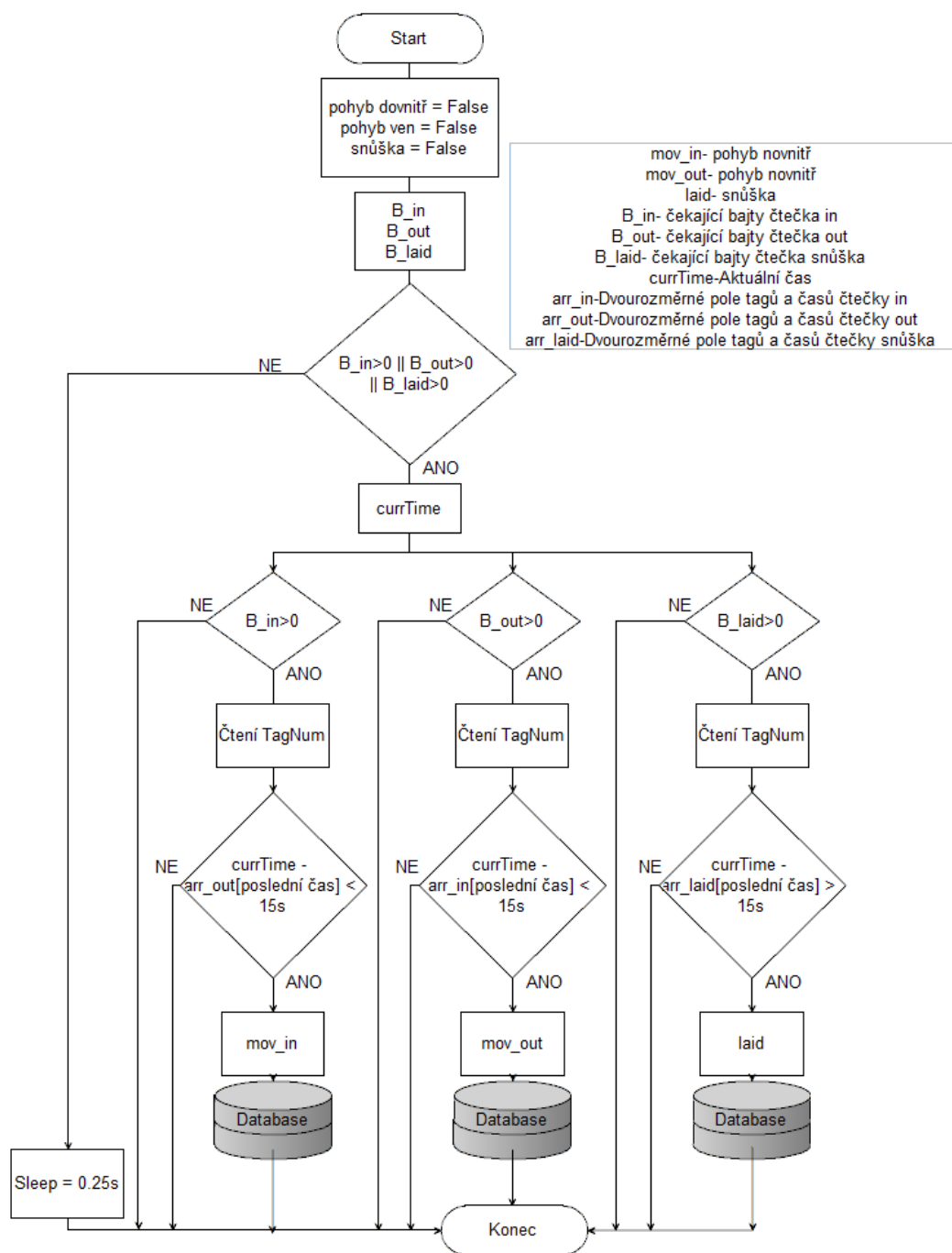
## **Funkce movementToDB**

Tato funkce slouží k zapsání pohybu do databáze. Její parametry jsou „tagNum“ – číslo tagu, „movTime“ – čas pohybu, „direction“ – směr pohybu a „typeMovement“ – typ pohybu, zda se jedná o pohyb vztažený ke kurníku nebo k místu snášení.

Prvním krokem je kontrola, zda číslu tagu z parametru „tagNum“ odpovídá nějaká nosnice v databázi. Pokud ano následuje druhý krok, kterým je vložení záznamu o pohybu do databáze. Typ pohybu je určen parametrem „typeMovement“.

## **Funkce separateTagNum**

Funkce „separateTagNum“ má pouze jeden parametr s názvem „data“ – data přečtená RFID čtečkou. Pomocí funkce „unpack\_from“ knihovny „struct“ je z dat přečtených čtečkou vyseparováno číslo RFID tagu, které je následně výstupem funkce „separateTagNum“.



Obrázek 6.8 Vývojový diagram monitorovacího algoritmu

## 7 Konfigurace a používání systému

Před zahájením sledování nosnic u kurníku, kde systém Sledování nosnic pomocí RFID není instalován, je nutno jej správně nakonfigurovat.

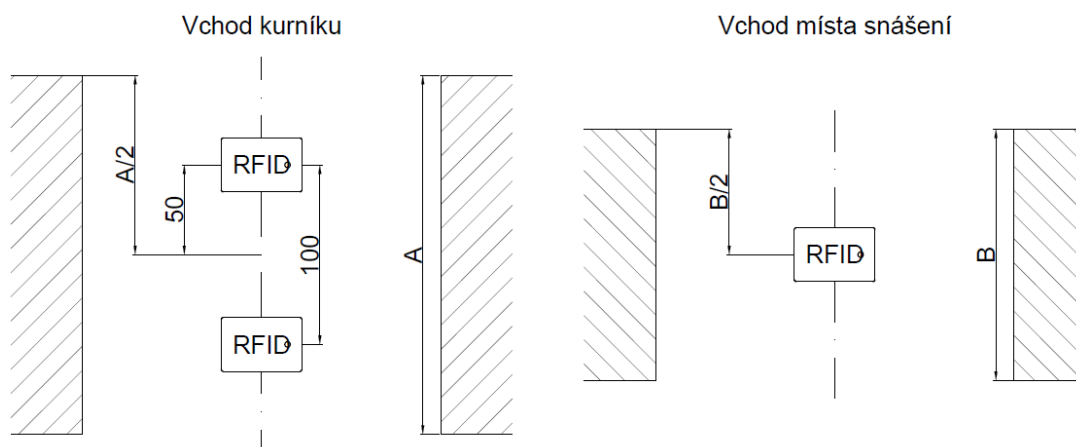
### Požadavky na kurník

Aby bylo možné do kurníku instalovat platformu Raspberry a byla zajištěna správná funkčnost, kurník musí splňovat tyto požadavky:

- Zásuvka se střídavým napětím 230 V
- Přístup k internetu pomocí Wi-Fi, případně po LAN kabelu
- Provozní teplota od 0°C do 50°C

### Instalace hardwarových prostředků

Čtečky určené pro vchod je třeba umístit do středu tohoto vchodu, na osu směru průchodu, aby docházelo ke spolehlivému čtení RFID tagů v obou směrech průchodu. Také je potřeba dodržet dostatečnou minimální vzdálenost čteček od sebe, aby při změně směru nosnice v průběhu procházení průchodem nedošlo k přečtení tagu oběma čtečkami současně. Tato vzdálenost by měla být zhruba 10 cm. Pozice čtečky pro průchod do místa snášení by měla být uprostřed tohoto průchodu. Všechny čtečky by měly být připevněny k podlaze tak, aby bylo zamezeno jejich pohybu.



Obrázek 7.1 Schéma rozmístění RFID čteček

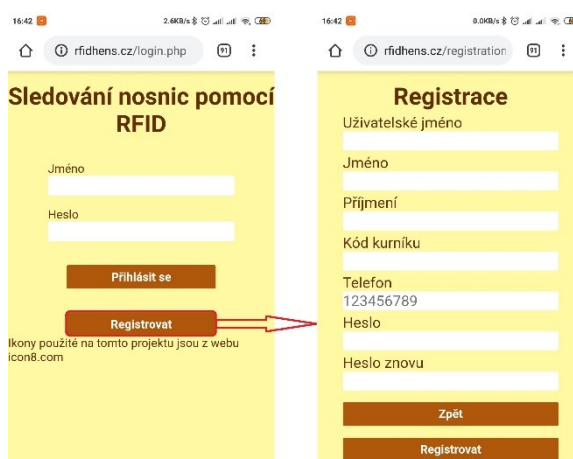
Umístění platformy Raspberry Pi musí být takové, aby kabely čteček vchodu do kurníku i vchodu do místa snášení k platformě dosáhly. Pokud to není možné, je nutné nedosahující kabely prodloužit prodlužovacím USB kabelem. Výška umístění, vzhledem k podlaze kurníku



by měla být taková, aby bylo zamezeno kontaktu nosnic s platformou, pro případ mechanického poškození oklováváním. V případě že takto platformu nelze umístit je nutné ji vhodně zakrytovat.

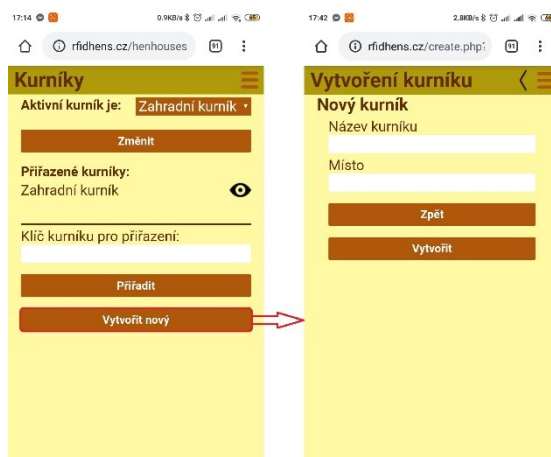
## Přidání kurníku ve webovém rozhraní

Webové rozhraní je dostupné na rfidhens.cz. Pro vstup do webového rozhraní je nutné přihlášení. Pokud pro přihlášení není dostupný uživatelský profil, je nutné jej vytvořit v sekci registrace. Tato sekce se zobrazí pomocí tlačítka „Registrovat“.



Obrázek 7.2 Stránky přihlášení a registrace

Po přihlášení do webového rozhraní je nový kurník přidán tak, že se na stránce „Kurníky“ stiskne tlačítko „Vytvořit nový“, což odkáže na stránku „Vytvoření kurníku“, kde se vyplní formulář a stiskne tlačítko „Vytvořit“. Nově vytvořený kurník je zobrazen na stránce „Kurníky“ v sekci „Přiřazené kurníky“.



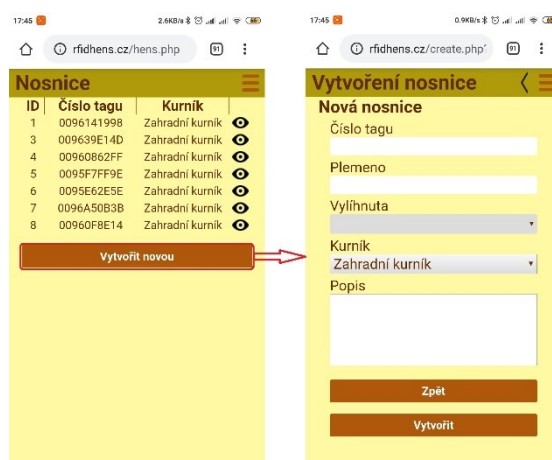
Obrázek 7.3 Stránky Kurníky a Vytvoření kurníku

## Nastavení platformy Raspberry Pi

Nejprve je k platformě Raspberry Pi připojen konektor USB C jejího napájecího zdroje, který je následně zapojen do zásuvky se střídavým napětím 230 V. Poté je připojen monitor k portu HDMI mini, klávesnici a myš k portům USB. Po načtení systému je nutné uložit soubor s algoritmem do uložistiště platformy. Následuje odpojení z USB portů klávesnice i myši a připojení čteček. Čtečky je nutné do hardwarové platformy Raspberry Pi zapojit ve správném pořadí, a to nejprve čtečku pro vnější stranu vchodu kurníku, následně pro vnitřní stranu vchodu a naposled čtečku pro místo snášení.

## Přidání nosnice a spuštění algoritmu pro monitorování nosnic

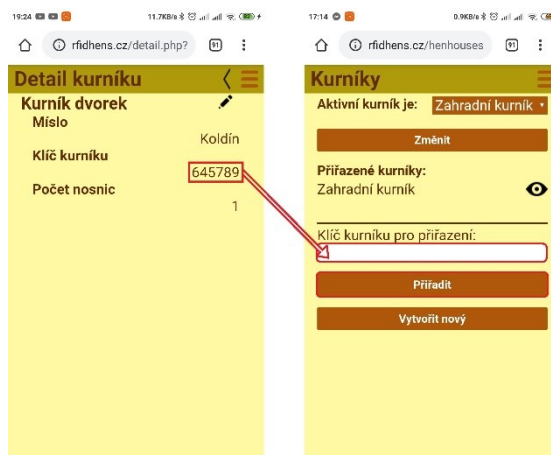
K volnému USB portu se připojí klávesnice a pomocí příkazu v příkazovém řádku se spustí soubor s algoritmem pro monitorování nosnic. Po spuštění je třeba vybrat z nabídky správný kurník, který lze jednoznačně identifikovat podle klíče kurníku. Dále je na výběr z možností „Přidat nosnici“ a „Spustit monitorování“. Při výběru „Spustit monitorování“ je spuštěn algoritmus pro monitorování nosnic. V případě volby „Přidat nosnici“ je nutné ke čtečce vnější strany vchodu kurníku přiložit RFID tag a podle výzev zapisovat údaje. Po vypsání všech údajů je nosnice přidána do databáze a znovu zobrazena nabídka „Přidat nosnici“ a „Spustit monitorování“. Přidání nové nosnice je také možné pomocí webového rozhraní, a to na stránce „Nosnice“ stisknutím tlačítka „Vytvořit novou“, vyplněním formuláře a potvrzením stisknutím tlačítka „Vytvořit“. Při přidávání nosnice přes webové rozhraní je nutné znát číslo RFID tagu nově přidávané nosnice.



Obrázek 7.4 Stránky Nosnice a Vytvoření nosice

## Přiřazení existujícího kurníku

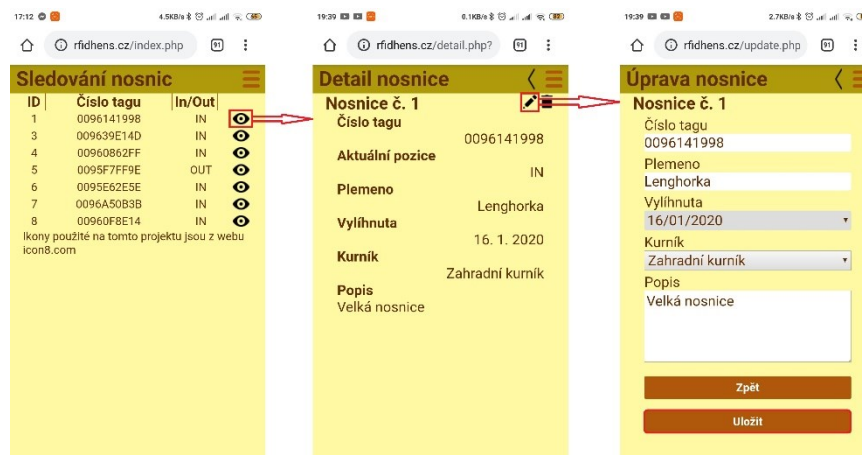
Přidání kurníku k uživatelskému účtu lze i jiným způsobem, než je vytvoření nového, a to přiřazením existujícího. K přiřazení kurníku je nutné znát jeho Klíč, který je u každého kurníku unikátní. Klíč kurníku je možné zjistit na stránce Detail kurníku. V dolní části stránky Kurníky je sekce obsahující formulář pro přiřazení kurníku pomocí „Klíče“, kde po zadání „Klíče“ a po stisknutí tlačítka „Přiřadit“ dojde k přiřazení kurníku k uživatelskému účtu.



Obrázek 7.5 Přidání kurníku pomocí Klíče kurníku

## Editace a detail

V případě, že došlo ke změně skutečností u uživatele, některé z nosnic nebo některého z kurníků, je možnost údaje editovat. Editační formulář se vždy zobrazí po stisknutí ikony „tužky“ na stránce Detail. Stránka detail je zobrazena stisknutím ikony „oka“. Po úpravě potřebných údajů v editačním formuláři jsou změny uloženy kliknutím na tlačítko Uložit.



Obrázek 7.6 Zobrazení detailu nosnice a editace jejích údajů

## 8 Rozšíření a další postup

Možná rozšíření systému do budoucna se nabízí ve vylepšení webového rozhraní, hardwaru i algoritmu sledování nosnic.

### Webové rozhraní a databáze

Hlavním rozšířením webového rozhraní by mohlo být přidání administrace, což by vedlo k rozšíření databáze. Dále by bylo možné připojit k uživatelskému účtu emailovou adresu a při registraci vyžadovat její ověření pomocí automaticky odesílaného emailu. Rozšíření v oblasti analýzy dat o pohybu a snášení nosnic by mohlo být vytváření grafů a zahrnovat do výsledků okolní teplotu nebo počasí.

### Hardware

Z hlediska hardwaru by se nabízelo rozšíření v podobě výměny stávajících RFID čteček za čtečky s externí anténou, kde by se dala předpokládat vyšší spolehlivost čtení. Také by bylo možné přidání technologie pro otevírání a zavírání dvířek kurníku. Tato technologie by byla ovládána upraveným algoritmem monitorovacího systému. Pro tento účel by byla vhodná technologie Kerbl a to z důvodu toho, že umožňuje externí ovládání přiváděním signálu na k tomu určený konektor.



Obrázek 8.1 Foto technologie Kerbl s vyznačeným konektorem pro externí ovládání, autor Ing. David Fojtík, Ph.D.

## 9 Závěr

Cílem práce bylo navrhnout a vytvořit systém pro monitorování nosnic. Nejprve byla vytvořena rešerše dostupných technologií inteligentního kurníku pro malochov, kde bylo zjištěno, že dostupné technologie se zaměřují pouze na otevírání a zavírání kurníku, přičemž neřeší skutečnost, zda jsou nosnice přítomny v kurníku nebo ne a přítomnost nosnic nijak nesledují. Zavírání a otevírání je u těchto technologií řízeno buď senzorem citlivým na světlo nebo pomocí časovače. Jako nejlepší z dostupných technologií bylo, po jejich vzájemném porovnání, vybráno řešení Kerbl, zejména proto, že umožňuje rozšíření o externí ovládání.

Na základě zjištěných skutečností při seznamování se s technologií RFID bylo zvoleno pracovní frekvenční pásmo nízkých frekvencí, které svými vlastnostmi postačuje. Z tohoto pásma byla vybrána pracovní frekvence 125kHz. Použité RFID tagy byly voleny s ohledem na pracovní frekvenci a požadovaný tvar kroužku, pro snadné umístění na končetinu nosnice. Rovněž RFID čtečky pro tento systém byly vybrány pro pracovní frekvenci 125kHz. Jako platforma monitorovacího systému slouží Raspberry Pi 4 Bodel B. Výhodami, které vedly k volbě této platformy, jsou dobrá konektivita (čtyři USB porty) a možnost vyvíjet algoritmus přímo na tomto zařízení.

Návrh webového rozhraní proběhl na základě předchozích zkušeností s tvorbou webů. Prvním krokem bylo navržení databáze, která byla v průběhu práce na webovém rozhraní upravována. Následoval grafický návrh rozložení elementů jednotlivých stránek, nejprve pro mobilní telefony a poté pro zařízení větších rozměrů. Vytváření webového rozhraní probíhalo na osobním počítači, kde bylo nutné vytvořit lokální server. Po otestování a odladění bylo webové rozhraní, pro možnost přístupu z internetu, nahráno na reálný server. Původně mělo být webové rozhraní nahráno na neplacený server, ale vyskytla se chyba při připojování k databázi z platformy Raspberry. Příčina této chyby byla na straně neplaceného serveru, který blokoval veškeré přístupy k databázi přicházející odjinud než z daného serveru. Z tohoto důvodu byla zakoupena doména [www.rfidhens.cz](http://www.rfidhens.cz) a rozhraní nahráno na placený server.

Algoritmus monitorovacího systému byl vyvíjen přímo na platformě Raspberry, ve vývojovém prostředí, které bylo předinstalováno v operačním systému platformy. Algoritmus slouží k vyhodnocování pohybů nosnic, ať už jde o pohyb ven nebo dovnitř kurníku, nebo pohyb v místě snášení. Záznamy o pohybech odesílá do databáze.

## 10 Použitá literatura a zdroje

- (1) Automatická dvířka chickdoor: Chytrý kurník. *Projekt Chytrý kurník: Chytrý kurník* [online]. [cit. 2020-11-28]. Dostupné z: [https://eshop.chytrykurnik.cz/automaticka\\_dvirka/automaticka-dvirka-chickdoor/](https://eshop.chytrykurnik.cz/automaticka_dvirka/automaticka-dvirka-chickdoor/)
- (2) SI59 - 12V automatický otvírač dveří. *MALAPA* [online]. 2020 [cit. 2019-12-06]. Dostupné z: <https://malapa.cz/automaticke-otevirani-kurniku/375-si59-12v-automaticky-otvirac-dveri.html>
- (3) Chicken Guard Premium. *Chicken Guard* [online]. Cambridge: ChickenGuard, 2020 [cit. 2020-01-10]. Dostupné z: <https://www.chickenguard.com/product/premium/>
- (4) Automatická dvířka kurníku KERBL. *Selko* [online]. 2020 [cit. 2020-05-20]. Dostupné z: <https://www.selko.cz/ostani-pomucky-a-prislusenstvi/7220-automaticka-dvirka-kurniku-kerbl-4018653705506.html>
- (5) LEHPAMER, Hrvoj. *RFID design principles*. 2nd ed. Boston: Artech House, 2012. ISBN 978-1-60807-470-9.
- (6) SOMMEROVÁ, Martina. *Základy RFID technologií: Výukový materiál*. Ostrava, 2003. Dostupné z: <https://docplayer.cz/3689532-Zaklady-rfid-technologie.html>
- (7) SMILEY, Suzanne. Active rfid vs passive rfid. *AtlasRFIDstore* [online]. Birmingham (Alabama): atlasRFIDstore.com, 2020 [cit. 2020-03-10]. Dostupné z: <https://www.atlasrfidstore.com/rfid-insider/active-rfid-vs-passive-rfid>
- (8) RFID Range: RFID Range Overview. *SkyRFID Inc.* [online]. Waterford, Canada: SkyRFID Inc., 2007-2019 [cit. 2019-12-11]. Dostupné z: [https://skyrfid.com/RFID\\_Range.php](https://skyrfid.com/RFID_Range.php)
- (9) MIKLÍKOVÁ, Marie. ID12-USB-SE-BOX: Ver 1.0 15.04.2016. *SOS electronic: ID12 - USB - SE - BOX* [online]. Místo: Nakladatelství, 2019 [cit. 2019-12-14]. Dostupné z: <https://cdn.sos.sk/productdata/e6/80/6fc8f5ac/id12-usb-se-box.pdf>
- (10) Raspberry Pi 4 Model B specifications – Raspberry Pi. In: *Raspberry Pi: Raspberry Pi 4* [pdf]. [cit. 2019-12-23]. Dostupné z: <https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-4-Product-Brief.pdf>
- (11) ANNANDALE, Greg. GPIO - Raspberry Pi Documentation. *Raspberry Pi* [online]. Cambridge: Raspberry Pi Foundation, 2019 [cit. 2019-12-03]. Dostupné z: <https://www.raspberrypi.org/documentation/usage/gpio/>
- (12) SUMMERFIELD, Mark. *Programming in Python 3: A Complete Introduction to the Python Language*. Second Edition. Boston: Addison-Wesley, 2010. ISBN 978-0321680563.
- (13) COWBURN, Peter, ed. What is PHP?. *PHP* [online]. PHP Documentation Group, 1997-2020 [cit. 2020-05-21]. Dostupné z: <https://www.php.net/manual/en/intro-whatis.php>
- (14) What is HTML (Hypertext Markup Language)?. *Computer Hope* [online]. 2020 [cit. 2020-05-24]. Dostupné z: <https://www.computerhope.com/jargon/h/html.htm>
- (15) What is CSS, How Does It Work and What is It Used For? - Skillcrush. *Skillcrush* [online]. 2012-2020 [cit. 2020-05-24]. Dostupné z: <https://skillcrush.com/blog/css/>

- (16) CATLIN, Hapton, Natalie WEIZENBAUM, Chris EPPSTEIN a Jina ANNE. Sass: Documentation. *Sass: Syntactically Awesome Style Sheets* [online]. 2006-2020 [cit. 2020-05-24]. Dostupné z: <https://sass-lang.com/documentation/>
- (17) BARNES, Russell. Thonny on a Raspberry Pi. In: *The MagPi Magazine: The official Raspberry Pi magazine* [online]. Raspberry Pi Trading, 2020 [cit. 2020-05-20]. Dostupné z: <https://magpi.raspberrypi.org/articles/thonny>
- (18) Documentation for Visual Studio Code. *Visual Studio Code* [online]. Seattle, 2020 [cit. 2020-05-18]. Dostupné z: <https://code.visualstudio.com/docs>